Processar fluxos de textos usando filtros

www.4linux.com.br

Sumário

Capítulo 1

Processar fluxos de textos usando filtros........3

1.1. Objetivos........3

1.2. Mãos a obra........4

Capítulo 2

Gerenciando 16

2.1. Objetivos........16

2.2. Troubleshooting 17

Índice de tabelas

Índice de Figuras

Capítulo 1

Processar fluxos de textos usando filtros

1.1. Objetivos

- Processar fluxos de textos;
- Utilizar filtros de saídas.

1.2. Mãos a obra

O comando cat é usado para mostrar o conteúdo de um arquivo, e pode atuar como um redirecionador de dados de sua entrada, enviando para uma saída. Vamos ver alguns exemplos práticos:

Exibir na saída padrão (tela) o conteúdo de um arquivo:

```
$ cat lista.txt
```

Como juntar 3 arquivos mp3 em apenas um só arquivo:

```
$ cat musica1.mp3 musica2.mp3 musica3.mp3 > musica.mp3
```

Um outro exemplo é enviar o conteúdo de um dispositivo para um arquivo .ISO:

```
# cat /dev/cdrom > CDDebian.iso
```

Usado para delimitar um arquivo em colunas, com a opção -d especifica qual o delimitador e -f quais campos serão exibidos. Exemplo:

Exibir o conteúdo das colunas 1 e 7 do arquivo /etc/passwd:

```
# cut -d ':' -f 1,7 /etc/passwd
```

Acrescentando ao final a opção --output-delimiter 'caractere' será exibido outro delimitador:

```
# cut -d ':' -f 1,7 /etc/passwd --output-delimiter '='
```

O comando Expand é usado para substituir espaços de tabulação por espaços simples, use a opção -t + o numero de linhas para selecionar a quantidade de espaços.

Exemplo de um aquivo texto com tabulação:

```
$ cat cadastro.txt
Nome		Cidade
Maria		Santos
João		Guarujá
José		Sao Paulo
```

Usando o comando expand vamos substituir os espaços de tabulação por apenas um espaço, redirecionando a saída para um novo arquivo:

```
$ expand -t1 cadastro.txt > novocadastro.txt
```

Exemplo de um aquivo texto após o comando expand:

```
$ cat novocadastro.txt
Nome Cidade
Maria Santos
João Guarujá
José Sao Paulo
```

O comando Unexpand é utilizado para substitui espaços simples por espaços de tabulação. O comando só terá efeito se o arquivo possuir dois espaços. Vamos ao exemplo:

```
$ cat carta.txt
 Esta  carta  pode  ser  usada  como  exemplo  para  o  comando  unexpand.
```

Exemplo do aquivo carta.txt texto após o uso do comando unexpand:

```
$ unexpand carta.txt
Esta		carta		pode		ser		usada		como
exemplo		para		o		comando		unexpand.
```

Fmt

Formata o conteúdo de um arquivo texto para um numero determinado de caracteres.

Exemplo da formatação de um arquivo para 60 caracteres:

```
$ fmt -w 60 /usr/share/doc/make/copyright
```

Head

Mostra as primeiras 10 linhas de um arquivo, e com -n você pode escolher a quantidade de linhas e -c a quantidade de caracteres:

Exemplo do comando para exibir apenas as 5 primeiras linhas de um arquivo:

```
$ head -n5 /usr/share/doc/make/copyright
```

Exemplo do comando para exibir apenas os 100 primeiros caracteres de um arquivo:

```
$ head -c100 /usr/share/doc/make/copyright
```

O Comando "Od – Octal Dump" pode ser usado para converter o conteúdo de um arquivo para diferentes formatos de dados. Opções do de formatos:

-c - ascII

-d - decimal

-f - vírgula flutuante

-o - octal

-x – hexadecimal

Veja alguns exemplos de conversão:

Conversão de um arquivo texto para hexadecimal:

```
$ od -x carta.txt
0000000 4520 7473 2061 6320 7261 6174 2020 6f70
0000020 6564 2020 6573 2072 7520 6173 6164 2020
0000040 6f63 6f6d 2020 7865 6d65 6c70 206f 7020
0000060 7261 2061 6f20 2020 6f63 616d 646e 206f
0000100 7520 656e 7078 6e61 2e64 000a
0000113
```

Conversão de um arquivo texto para vírgula flutuante:

```
$ od -f carta.txt
0000000   7,709535e+31   2,953812e+21   2,818280e+20   7,431524e+28
0000020   1,358607e-19   2,053176e-19   2,806874e+20   1,358475e-19
0000040   7,348261e+28   1,858887e+34   1,162635e+27   1,982272e+29
0000060   1,909609e-19   1,358930e-19   2,737440e+20   2,027732e-19
0000100   7,038024e+22   1,744252e+28   9,349968e-40
0000113
```

O comando Join exibe o conteúdo de arquivos lado a lado usando como base um campo correspondente

Em nosso exemplo sera usado 2 arquivos, o lista1 e lista2:

```
$ cat lista1
1	Maria
2	João
3	José
```

```
$ cat lista2
1	18 anos
2	23 anos
3	31 anos
```

Para juntar use o comando join com a opção -j e o valor do campo correspondente nos dois arquivos:

```
$ join -j 1  lista1 lista2
1 Maria 18 anos
2 João 23 anos
3 José 31 anos
```

O comando nl numera linhas não vazias, com -ba numera todas as linhas.

Exemplo do comando para numerar linhas não vazias:

```
$ nl videogames
1  Playstation

2  X-box
3  Sega Genesis
4  Master System

5  Atari
```

```
6  Nintendo
7  Super Nintendo
```

Exemplo do comando para numerar todas as linhas:

```
$ nl videogames
1  Playstation
2
3  X-box
4  Sega Genesis
5
6  Master System
7  Atari
8
9  Nintendo
10  Super Nintendo
```

Paste

Comano usado para exibir o conteúdo de arquivo lado a lado em forma de colunas, com a opção -s em forma de linhas, e usado com -d 'caractere' acrescenta um delimitador.

Em nosso exemplo serão usados os arquivos usuarios e provedores:

```
$ cat usuarios
maria
joao
jose
```

```
$ cat provedores
terra.com.br
bol.com.br
uol.com.br
```

Para juntar o conteúdo dos arquivos lado a lado:

```
$ paste usuarios provedores
maria   terra.com.br
joao    bol.com.br
jose    uol.com.br
```

Para juntar o conteúdo dos arquivos e exibir em linhas:

```
$ paste -s usuarios provedores
maria   joao    jose
terra.com.br    bol.com.br      uol.com.br
```

Para juntar o conteúdo dos arquivos lado a lado usando um delimitador:

```
$ paste -d@ usuarios provedores
maria@terra.com.br
joao@bol.com.br
jose@uol.com.br
```

Pr

Divide o arquivo para impressão, com a opção -w define a quantidade de caracteres:

```
$ pr -w 60 /usr/share/doc/make/copyright
```

Sed

Sed é um stream editor, usado para varias tarefas no terminal, como por exemplo procurar e substituir padrões em textos mostrando o resultado em stdout

opções

-e - Executa a expressão e comando a seguir;

-f - Lê expressões e comandos do arquivo indicado pela opção;

-n - Não Mostra as linhas que não correspondam à expressão.

Comandos comuns de sed:

s – Substituir;

d - Apaga a linha;

r - Insere o conteúdo do arquivo indicado na ocorrência da expressão;

w - Escreve a saída no arquivo indicado;

g - Substitui todas as ocorrências da expressão na linha atual.

Veja alguns exemplos de uso do comando sed:

Exibir a substituição do conteúdo de arquivo:

Sintaxe:

sed "s/string_original/string_troca/g" arquivo_original

```
# sed “s/bash/sh/g” /etc/passwd
```

Enviar a substituição do conteúdo do arquivo para outro arquivo:

```
# sed “s/bash/sh/g” /etc/passwd > /root/newpasswd
```

Faz a substituição do conteúdo do arquivo no próprio arquivo original:

```
# sed -i “s/bash/sh/g” /root/passwd
```

Type

Comando usado para exibir a interpretação de outros comandos. Com a opção -t podemos saber se o comando é um alias, interno ou um arquivo binario. Veja alguns exemplos:

Interpretação do comando date:

```
# type -t date
file
```

Interpretação do comando pwd:

```
# type -t pwd
builtin
```

Interpretação do comando lt:

```
# type -t pwd
alias
```

Split

Divide um arquivo em varias partes, com -l pela quantidade de linhas e -b pela quantidade de kbytes.

Como exemplo sera usado um arquivo de texto de 8 linhas e vamos dividir em 2 arquivos de 4 linhas cada.

Exemplo do arquivo original com todas as linhas:

```
# cat videogames
Playstation
X-box
Sega Genesis
Master System
Atari
Nintendo
Super Nintendo
Jaguar
```

Exemplo da divisão em 2 arquivos:

```
# split -l 4 videogames parte_
```

Conteúdo do primeiro arquivo:

```
# cat parte_aa
Playstation
X-box
Sega Genesis
Master System
```

Conteúdo do segundo arquivo:

```
# cat parte_ab
Atari
Nintendo
Super Nintendo
Jaguar
```

Tail

Mostra as ultimas 10 linhas de um arquivo. Com a opção -n você pode escolher a quantidade de linhas exibidas, com -c a quantidade de caracteres e com -f é mostrado continuamente em tempo real.

Exemplo do comando para exibir as 10 ultimas linhas de um arquivo:

```
# tail /usr/share/doc/make/copyright
```

Exemplo do comando para exibir continuamente as ultimas linhas de um arquivo de log:

```
# tail -f /var/log/auth.log
```

Tr

Comando usado para converter caracteres do conteúdo de arquivos.

Exemplo do comando para converter letras minusculas por letras maiúsculas:

```
# tr '[a-z]' '[A-Z]' < videogames
PLAYSTATION
X-BOX
SEGA GENESIS
MASTER SYSTEM
ATARI
NINTENDO
SUPER NINTENDO
JAGUAR
```

Uniq

Mostra o conteúdo de arquivos ocultando linhas repetidas, com -u mostra apenas linhas que não se repetem e com a opção -d as linhas repetidas. Vamos aos exemplos:

Exibir linhas ocultando as repetidas:

```
# uniq videogames
Playstation
X-box
Sega Genesis
Master System
Atari
Nintendo
Super Nintendo
Jaguar
```

Exibir linhas que não são repetidas:

```
# uniq -u videogames
Playstation
Sega Genesis
Master System
Atari
Super Nintendo
Jaguar
```

Exibir linhas que são repetidas:

```
# uniq -d videogames
X-box
Nintendo
```

Wc

Comando usado para conta linhas, palavras e caracteres. Opções do comando:

-l – linhas;

w – palavras;

-c – caracteres.

Conta quantas linhas o arquivo possui:

```
# wc -l videogames
10 videogames
```

Capítulo 2

Gerenciando

2.1. Objetivos

- Resolução de problemas utilizando comandos.

2.2. Troubleshooting

Como faço para remover todos os comentários e linhas em branco de arquivos de configurações?

Através do comando sed é possível realizar essas tarefas e jogar a saida para um outo arquivo. Vamos aos exemplos:

Remover todos os comentários do arquivo de configuração do samba e enviar para um novo arquivo:

```
# sed '/^#/d' /etc/samba/smb.conf > /root/smb_no_comments
```

Remover todas as linhas em branco do arquivo de configuração do samba e enviar para um novo arquivo:

```
# sed '/^$/d' /etc/samba/smb.conf > /root/smb_no_blank_lines
```